# Development of Extended XDB Protocol and Prototype

Wook-Sung Yoo[+], Spoorthy Gowda and Vishwanath Mamillapalli

Fairfield University, Software Engineering Program, 1073 North Benson Road, Fairfield, CT 06824, USA

**Abstract.** XDB is an open-source and extensible database architecture, developed by NASA (National Aeronautics and Space Administration), to provide seamless integration of heterogeneous and distributed information resources for scientific and engineering applications. XDB enables unlimited number of desktops and distributed information sources to be linked seamlessly and efficiently into an information grid using Data Access and Retrieval Composition (DARC) protocol which provides a contextual search and retrieval capability useful for lightweight web applications. Supported by NASA Exploration System Mission Directorate (ESMD) Higher Education, capstone project team at Fairfield University created an extended XDB protocol and a prototype providing text-searches for Wiki. This paper describes the prototype of extended XDB demonstrating the usage of XDB in data management without burdening users with complex formal schemas. The prototype has been created for sixteen tags of the Mediawiki dialect. As future works, the prototype will be extended to the complete set of Mediawiki markups and other dialects of wiki.

**Keywords:** NSAS, Database, XDB, DARC, WebDAV, Wiki

## 1. Introduction

NASA (National Aeronautics and Space Administration) engineering teams have struggled to access existing information in databases and files with hundreds different formats in their own explicit and implicit structures. The decision making applications to access this information are required to follow numerous procedures, guidelines, and complex work practices. To resolve this problem, XDB, an open-source and extensible database architecture, was developed by NASA to provide seamless integration of these heterogeneous and distributed information resources for scientific and engineering applications [1]. XDB provides a novel "schema-less" database approach using a document-centered object-relational XML database mapping [2], [3]. This enables structured, unstructured, and semi-structured information to be integrated without requiring document schemas or translation tables. XDB utilizes existing international protocol standards of the World Wide Web Consortium Architecture Domain and the Internet Engineering Task Force, primarily HTTP, XML, and WebDAV (Web Distributed Authoring and Versioning), an extension of the Hypertext Transfer Protocol (HTTP) that facilitates collaboration between users in editing and managing documents and files stored on World Wide Web servers [4]. Through a combination of these international protocols, universal database record identifiers, and physical address data types, XDB enables an unlimited number of desktops and distributed information sources to be linked seamlessly and efficiently into an information grid [5]. XDB uses Data Access and Retrieval Composition protocol, a REST HTTP query protocol for retrieving and recomposing XML and HTML documents stored on a remote server [6]. DARC provides a contextual search and retrieval capability useful for lightweight web applications such as AJAX or PHP middleware that removes the need of schemas and schema management while providing the capability to retrieve relevant data from large, semi-structured stores. As a part of the NASA Exploration System Mission Directorate's Higher Education Project, a draft of the XDB protocol and source code were provided to capstone project team in software engineering program at Fairfield University to create a prototype to prove the concept. Teamed with NASA Ames Research Center, technical specification of the protocol was updated to SourceForge and a prototype providing text-searches for Wiki was developed.

## 2. Methodology

As described in Fig. 1, the system mainly consists of Web interface with APIs, XDB Database, and XML parser. The keywords for context and content searches can be entered and a search phrase is created. All

[+] Corresponding author. Tel.: + 001-203-254-4000; fax: + 203-254-4013.
*E-mail address*: wyoo@fairfield.edu

queries contain one or more expressions, separated with the ampersand ("&") character. As an example, the query for name "Richard" should be written as http://localhost/DARC?context=Name &content="Richard." After system searches and parses the documents, contexts that match any of the expressions are returned and displayed in XML format on the web browser.

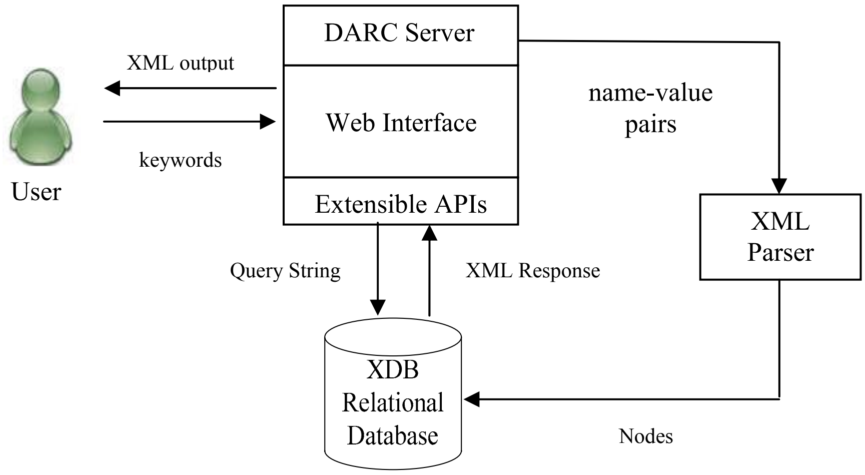


Fig. 1: XDB System Overview

Current XDB protocol was defined and implemented for documents in XML and HTML formats and we selected Wiki, a lightweight mark-up language, to develop a prototype using XDB as a reference implementation. The prototype of Wiki version stores information of resources in heterogeneous formats, organizes resources in hierarchy, and allows context and content querying of information in the XML database

## 2.1. Specification of DARC

The Data Access and Retrieval Composition (DARC) protocol provides contextual search and retrieval of semi-structured data. Contextual search is searching for data based on where it is located in the structure, primarily focused on the immediate context of the information sought. For example, if a user wishes to find their name in the "Author" section of a set of documents, the search can be quickly performed with no fore-knowledge of the structure of the documents being searched. Contextual retrieval adds the capability to retrieve only the relevant section of the documents that match the query expression, which reduces the network burden required to find the relevant information and provides further analysis function. Following the previous example of "Author", the user's query returns the documents containing the name in the "Author" section in a structured output. The user could then store the output for further analysis.

DARC is a protocol that follows the Representational State Transfer (REST) architecture. DARC queries are performed as HTTP "GET" or "POST", and DARC query results returned in the HTTP response in XML or ASCII. Queries are read-only, stateless operations on the DARC data store. DARC uses standard HTTP requests, caching, proxying, authentication and access control, encryption, compression and other HTTP-related technologies. To update data in a DARC index, WebDAV was used to leverage the simplicity of remote data management. DARC relies on documents having Unique Resource Identifiers both in the query expressions and in the result output when it identifies matches. DARC is aware of any updates to the underlying resources and each URI uniquely identifies one and only one resource as per the semantics of the URI standard. All DARC queries, as Restful operations, require a base URI. All query expressions and options are specified as part of the "query string" of the URL and the query response is an XML HTTP response body so that users can interact with a DARC system directly from within their browser by typing queries into the URL field and examining the structured XML in the page view of the browser. For an example, a GET request of http://publib.library.org/darc?author=Chris%20Knight&syntax=xml would return:

```
<?xml version="1.0"?>
<resultset>
  <result>
```

```
        <meta><uri>/specifications/darc.xml</uri></meta>
        <value><name>Richard</name></value>
    </result>
    </resultset>
```

DARC servers return identical results to the expressions which are syntactically different but semantically equivalent.

## 2.2. Prototype Development with Wiki

The content and context search feature in Wiki prototype enables an end user to retrieve information from highly complex and constantly changing heterogeneous data formats into a well-structured, common standard so that the end-user can index documents in Wiki text format by running a single command [7].

As described in Fig. 2, the prototype handles the query with two main phases: (1) wiki to xml conversion using sgml and (2) indexing the converted xml in the database. Perl script is used to insert converted document to database.

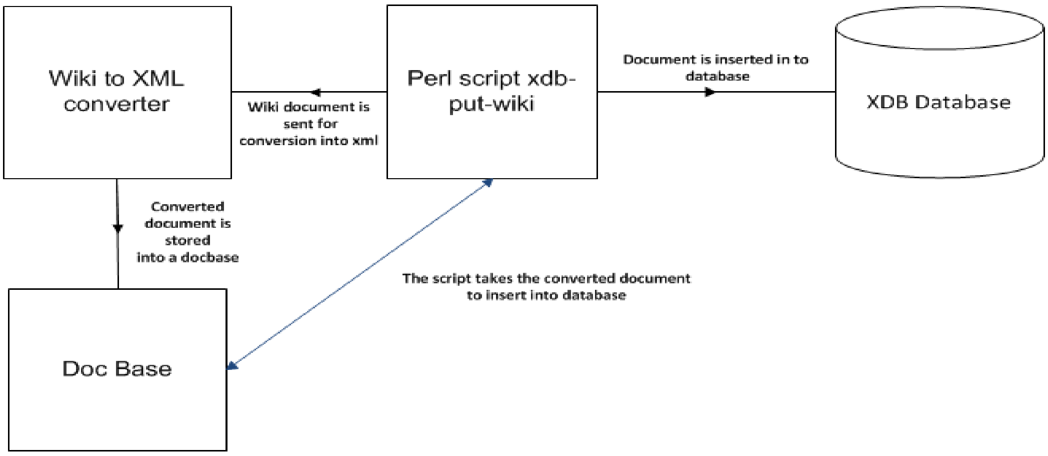

Wiki to XML converter
Wiki document is sent for conversion into xml
Perl script xdb-put-wiki
Document is inserted in to database
XDB Database
Converted document is stored into a docbase
The script takes the converted document to insert into database
Doc Base

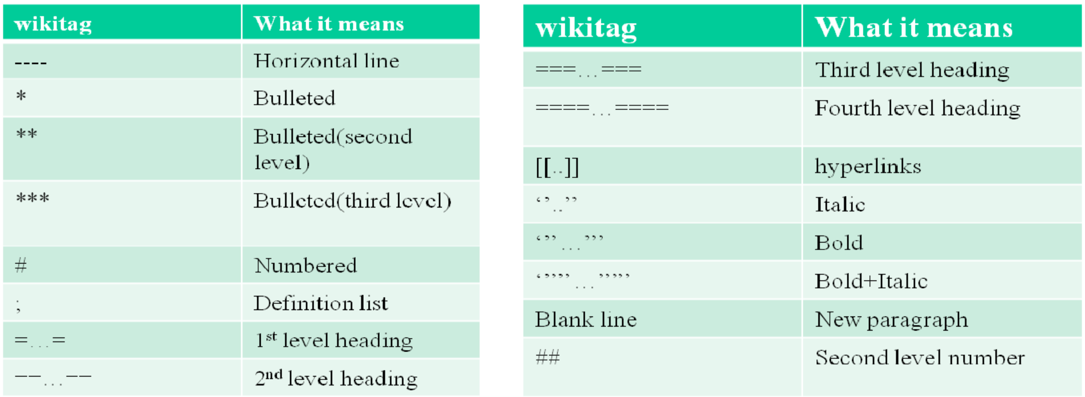
Fig. 2: Diagram for the Prototype

Wiki to XML Converter Module has two sub modules: SX (SGML to XML) and Wiki.spm (DTD).

SX, a free object-oriented toolkit for SGML parsing and entity management, converts SGML to XML. SX parses and validates the SGML document and writes an equivalent XML document to the standard output. SX will warn about SGML constructs which have no XML equivalent. SX takes the following arguments:

*sx [ -Cehilprvx ] [ -bencoding ] [ -ccatalog_file ] [ -Ddirectory ] [ -ffile ] [ -wwarning_type ] [ -xxml_output_option ] sysid...*

Wiki.sgml (DTD) Module converts Wiki to SGML. DTD in SGML is used to convert Wiki to SGML by defining character set including start and ending tags and mapping wikitags to corresponding XM. Fig. 3 shows the 16 tags converted.



| wikitag | What it means |
|---|---|
| ---- | Horizontal line |
| * | Bulleted |
| ** | Bulleted(second level) |
| *** | Bulleted(third level) |
| # | Numbered |
| ; | Definition list |
| =...= | 1st level heading |
| ==...== | 2nd level heading |

| wikitag | What it means |
|---|---|
| ===...=== | Third level heading |
| ====...==== | Fourth level heading |
| [[..]] | hyperlinks |
| ‘’..’’ | Italic |
| ‘’’...’’’ | Bold |
| ‘’’’’...’’’’’ | Bold+Italic |
| Blank line | New paragraph |
| ## | Second level number |

Fig. 3: Converting Wiki to XML in 16 tags

Perl script, xdb-put-wiki, is created to put searched document to XDB database:

*xdb-put-wiki [database name] [URI] [filepath]*

The format is consistent with xdb-put-xml and xdb-put-html. All source files in /test/data and all error logs is saved in /error directory.

## 3. Conclusion

The prototype was successfully developed and source code was posted as open source at Source Forge (www.sourceforge.net). The draft of protocol was examined and changes were made to the specification document based on the testing. Conversion process of the specification document to mediawiki is also posted at sourceforge and changes shared via wikipage. The SGML DTD currently supports 16 main tags (mediawiki). Future development includes development of all tags of mediawiki and extending to other dialects of Wiki.

## 4. Acknowledgements

This research was financially supported by the National Space Grant Foundation (NASA Technology Database: AERC2-05-SD) at Fairfield University.

## 5. References

[1] D. A. Maluf, P. B. Tran, and Y. Gawdiak, New Information System Provides Multiple Data Source Access. *NASA Ames Astrogram Article*, October 2002.

[2] D. A. Maluf and P. B. Tran, Articulation Management for Intelligent Integration of Information. *IEEE Transactions on Systems, Man, and Cybernetics*, November 2001, vol. 31, No. 4, pp. 485-496.

[3] D. A. Maluf, D. G. Bell, C. Knight1, P. Tran, T. La, J. Lin, B. McDermott, and B. Pell, XDB-IPG: An Extensible Database Architecture for an Information Grid of Heterogeneous and Distributed Information Resources. Stanford Article, *Information Management*: XDB-IPG-0.9.

[4] E. J. Whitehead, and Y. Goland, The WebDAV Property Design. *Software, Practice and Experience*, 2004, 34, pp.135-161.

[5] NASA Information Power Grid (IPG). http://www.ipg.nasa.gov/.

[6] E. R. Harold, XML: Extensible Markup Language. *IDG Books Worldwide*, 1998.

[7] D. Aumueller, Semantic authoring and retrieval within a Wiki. *Proceeding of Second ESWC*, 2005